REMARKS

In the Official Action mailed on **2 April 2008**, the Examiner reviewed claims 1-31. Examiner rejected claims 2 and 19-20 under 35 U.S.C. § 112. Examiner rejected claims 1, 6-16, 18, 20, 23-24, and 30-31 under 35 U.S.C. § 102(b) based on DeLong et al. (U.S. Patent No. 5,892,947, hereinafter "DeLong"). Examiner rejected claims 2-5, 17, 19, 21-22, and 25-29 under 35 U.S.C. § 103(a) based on DeLong and Holzman et al. (U.S. Patent No. 6,804,634, hereinafter "Holzman").

Rejections under 35 U.S.C. § 112

Examiner rejected claims 2, 19, and 20 under U.S.C. § 112, second paragraph, as being indefinite, averring that the limitations *completely* and *effectively* are not clear and concise. Applicant respectfully notes that the term *completely* corresponds to a *code coverage* rating (see instant application, page 7, lines 3-11), and that the term *effectively* corresponds to an ability of a test case to locate certain types of bugs (see instant application, page 8, lines 7-12). Accordingly, applicant has amended claims 2 and 20 to clarify that embodiments of the present invention calculate a coverage rating to indicate the *coverage achieved* from executing said test case on said software component. Applicant has also amended claim 19 to clarify that a rating is configured to indicate the ability of a test case to locate bugs on the software component. These amendments find support on page 7, lines 3-11, and page 8, lines 7-12 of the instant application. No new matter has been added.

Rejections under 35 U.S.C. § 102 and § 103

Examiner has rejected claims 1, 6-16, 18, 20, 23-24, and 30-31 under 35 U.S.C. § 102(b) based on DeLong, and Examiner has rejected claims 2-5, 17, 19,

21-22, and 25-29 under U.S.C. § 103(a) as being unpatentable over DeLong in view of Holzman. Applicant respectfully disagrees, because combining DeLong with Holzman does not teach calculating a coverage rating configured to indicate the coverage achieved from **executing** a test case on a software component, wherein the **coverage rating is normalized to exclude dead code**.

Specifically, Holzman discloses generating and regenerating coverage test case sets directly from a model (see Holzman, col. 2, lines 36-39). The user builds a model to represent the test requirements of a software component, where the model is a finite state machine (FSM) or hierarchical graph (see Holzman, col. 1, lines 18-20, and col. 2, lines 39-42). A model is commonly known in the art to represent the target functionality of a system, which does not include dead code because dead code is non-functional. Therefore, because a model in the Holzman system does not include dead code, the Holzman system cannot be combined with the DeLong system to provide a coverage rating that ignores the effect of dead code; the Holzman system **can only provide a coverage rating as to what can be covered**, since it produces a coverage rating based on the model only. Holzman only discloses details on coverage with respect to traversing the model's graph to generate test cases that can achieve a coverage goal specified by a user (see Holzman, col. 2, lines 43-51). Applicant respectfully notes that the presence of dead code in a software component would cause their maximum coverage rating possible for the software component to be below 100%. Such a scenario does not provide a user with a guarantee that he has stimulated the complete software functionality, because the user is attempting to achieve 100% coverage.

In contrast, embodiments of the present invention generate a coverage rating by **observing execution of test cases** and **determining how much of a software component is exercised by each test case** (see instant application, page 7, lines 8-11). The code coverage rating reflects a percentage of all code in a source file that is tested by a test case, without considering dead code in the rating.

In other words, the code coverage ratings provided by embodiments of the present invention **are normalized to exclude dead code.** (see instant application, page 7, lines 3-19). Excluding the presence of dead code in a coverage rating is an important feature of the coverage rating because it provides a more accurate rating, where a coverage rating less than 100% **signifies that *executable* code remains that has not been tested**.

DeLong and Holzman do not disclose, either explicitly or implicitly, either alone or in concert, calculating a coverage rating configured to indicate the coverage achieved from **executing** a test case on a software component, wherein the coverage rating **is normalized to exclude dead code**.

Accordingly, applicant has amended independent claims 1, 17, 18, and 23 to further clarify that embodiments of the present invention **calculate a coverage rating** configured to indicate the coverage **achieved from executing a test case** on a software component, **wherein the coverage rating is normalized to exclude dead code**. These amendments are supported by original claims 2, 20, and 24, and by page 7, lines 3-19, and page 11, lines 9-15 of the instant application. Applicant has canceled dependent claims 2, 20, and 24. No new matter has been added.

Hence, Applicant respectfully submits that independent claims 1, 17, 18, and 23 as presently amended are in condition for allowance. Applicant also submits that claims 2-16, which depend upon claim 1, claims 19-23, which depend upon claim 18, and claims 24-31, which depend upon claim 23, are for the same reasons in condition for allowance and for reasons of the unique combinations recited in such claims.

CONCLUSION

It is submitted that the application is presently in form for allowance. Such action is respectfully requested.

Respectfully submitted,

By /Anthony Jones/
Anthony Jones
Registration No. 59,521

Date: 30 June 2008

Anthony Jones
Park, Vaughan & Fleming LLP
2820 Fifth Street
Davis, CA 95618-7759
Tel: (530) 759-1666
Fax: (530) 759-1665
Email: tony@parklegal.com